



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/639,674	08/11/2003	Eric Moore	11973-007001	6822
26181	7590	06/15/2007	EXAMINER	
FISH & RICHARDSON P.C. PO BOX 1022 MINNEAPOLIS, MN 55440-1022			WEI, ZHENG	
		ART UNIT	PAPER NUMBER	
		2192		
		MAIL DATE		DELIVERY MODE
		06/15/2007		PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

<b>Office Action Summary</b>	<b>Application No.</b>	<b>Applicant(s)</b>
	10/639,674	MOORE ET AL.
	<b>Examiner</b>	<b>Art Unit</b>
	Zheng Wei	2192

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

#### Status

- 1) Responsive to communication(s) filed on 12 March 2007 and 13 April 2007.  
 2a) This action is FINAL.                    2b) This action is non-final.  
 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

#### Disposition of Claims

- 4) Claim(s) 1-29 is/are pending in the application.  
 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.  
 5) Claim(s) \_\_\_\_\_ is/are allowed.  
 6) Claim(s) 1-29 is/are rejected.  
 7) Claim(s) \_\_\_\_\_ is/are objected to.  
 8) Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

#### Application Papers

- 9) The specification is objected to by the Examiner.  
 10) The drawing(s) filed on 11 August 2003 is/are: a) accepted or b) objected to by the Examiner.  
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).  
 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

#### Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).  
 a) All    b) Some \* c) None of:  
 1. Certified copies of the priority documents have been received.  
 2. Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.  
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

#### Attachment(s)

- |  |   |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892)                     | 4) <input type="checkbox"/> Interview Summary (PTO-413)           |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____                                      |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)          | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____. _____   | 6) <input type="checkbox"/> Other: _____                          |

## **DETAILED ACTION**

### ***Remarks***

1. This office action is in response to the amendment filed on 03/12/2007.
2. Claims 1, 3, 13, 16 and 26 have been amended.
3. The objection to the specification is withdrawn in view of Applicant's amendment
4. The objection to claim 3 is withdrawn in view of amended claim.
5. Claims 1-29 remain pending and have been examined.

### ***Oath/Declaration***

6. The Oath/Declaration filed on 04/13/2006 has been accepted and put in the application file. Therefore, the objection is withdrawn.

### ***Response to Amendment***

7. Applicant's amendment filed on 03/12/2007, changes the scope of claims 1-29. Therefore, a new ground of rejection is applied.

### ***Response to Arguments***

8. Applicant's arguments filed on 03/12/2007, in particular on pages 8-10, has been fully considered but they are not persuasive. For example:

- At page 8, section 102 rejections, third paragraph, Applicant contends that the cited portions of Highland disclose translating rules of the knowledge base into a rule tree network and each node in the rule tree is a rule component. The Examiner disagrees. First of all, the data structure “tree” as defined in Wikipedia is a type of graph (see for example, <[http://en.wikipedia.org/wiki/Tree\\_data\\_structure](http://en.wikipedia.org/wiki/Tree_data_structure)>, “In graph theory, a tree is a connected acyclic graph. A rooted tree is such a graph with a vertex singled out as the root. In this case, any two vertices connected by an edge inherit a parent-child relationship.”). Moreover, Highland discloses in Fig.1B (see for example, node “goal”, node “Action”, node condition (“y<10”)) and in the specification (see for example, col.6, lines 28-36, about “sub tree” which the root of sub tree is equivalent to the attribute node) about the ranked nodes. Therefore, the examiner reasserted that Highland, indeed, anticipated the claimed limitations as set forth in the previous Office action and amended claims.
- At page 9, last paragraph, Applicant points out that “The examiner” states “Burke discloses determining and resolving logical conflicts at col.1, lines 59-64”. However, the Examiner has never cited above portion. As previous office action, what the Examiner cited is col.3, lines 30-37.
- At page 10, section 103 rejection, For the same reasons set forth above with respect to the method claims 1, the reasserted that Highland also anticipated

the claimed limitations as set forth in the previous Office action and amended claims.

***Claim Rejections - 35 USC § 102***

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

10. Claims 1, 2, 5 and 6 are rejected under 35 U.S.C. 102(b) as being anticipated by Highland (Frederic D. Highland, US 4,924,408)

Claim 1:

Highland discloses a method comprising:

- in a processing system, receiving a rule set as a single package (see for example, col.1, lines 51-52, "knowledge base");
- generating a dependency graph for the rule set, the dependency graph including a plurality of ranked nodes, the nodes including entity nodes, attribute nodes, condition nodes, and rule nodes (see for example, node

"goal", node "Action", node condition ("y<10") and col.6, lines 28-36, about "sub tree" which the root of sub tree is equivalent to the attribute node; also see Fig.2a-2b and related text; further see Col.5, lines 50-55, Brief Description of the Drawings about Fig.2); and

- generating a sequence of processing logic for optimal processing of inputted facts according to a rank order of the nodes in the dependency graph (see for example, Fig.2a-2c and related text, also see Col.5, lines 50-55, Brief Description of the Drawings about Fig.2).

Claim 2:

Highland further discloses the method of claim 1, in which processing comprises single pass execution when there are no logical loops (see for example, col.2, lines 52-55, "The user of a tree structure guarantees that only one parent exists for each node in the rule tree, eliminating the need for multiway branching...").

Claim 5:

Highland also discloses the method of claim 1 in which the rule set is free of logical conflicts (see for example, col.1, lines 59-64, "...the actions to be performed when a rule is 'true'").

Claim 6:

Highland further discloses the method of claim 1 in which generating the dependency graph comprises determining logical dependencies across rules contained in the rule set (see for example, col.6, lines 21-36, "Step 0 and Step 1" and related descriptions).

11. Claims 16-18, 20 and 25 are rejected under 35 U.S.C. 102(b) as being anticipated by Burke (Burke et al., US 5,423,041)

Claim 16:

Burke discloses a method for automating business processes comprising:

- in a computer system, receiving a rule set as a single package (see for example, col.3, line 30, "Sets of rules are organized as rule-sets");
- determining logical conflicts within the rule set where a logical conflict exists when two or more rules receiving the same inputs result in contradictory actions (see for example, col.3, lines 33-34, "If multiple rules are satisfied at the same time...");
- resolving the logical conflicts (see for example, col.3, lines 32-35, "Within a rule-set, an application programmer may assign each rule a priority. If multiple rules are satisfied at the same time, the rule with highest priority is selected for firing."); and
- generating a sequence of processing logic from the rule set for optimal processing of inputted facts using the resolved logical conflicts (see for

example, col.3, lines 36-37, "The inference engine of the rules system executes the match, select, and act phases of the inferencing process.").

Claim 17:

Burke further discloses the method of claim 16, in which resolving comprises determining override conditions in rule collision events (see for example, col.3, lines 32-35, "Within a rule-set, an application programmer may assign each rule a priority. If multiple rules are satisfied at the same time, the rule with highest priority is selected for firing.").

Claim 18:

Burke also discloses the method of claim 16, in which generating comprises analyzing the rule set with a business logic generation utility optimized for one of a plurality of target programming languages and generating optimized business logic for the selected target programming language (see for example, col.4, lines 22-41, "Step 11 is coding a system of rules", "In step 13...This translation step is accomplished with a rules compiler" and "In step 14, the rules code is compiled with a standard C++ compiler to generate object code").

Claim 20:

Burke also discloses the method of claim 18, in which the target programming language is C++(see for example, col.4, lines 40-41, "In step 14, the rules code is compiled with a standard C++ compiler to generate object code").

Claim 25:

Burke further discloses the method of claim 18, in which the business logic generation utility's generated processing logic comprises a series of calls to a working memory database to retrieve, manipulate and update data (see for example, col.3, lines 59-61, "During its match phase, the inference engine tests each rule's premise against the current working memory", also see col.4, lines 10-12, "These actions may update the working memory by creating new objects and removing old objects.").

***Claim Rejections - 35 USC § 103***

12. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:
  - (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.
13. Claims 3, 4 are rejected under 35 U.S.C. 103(a) as being unpatentable over Highland (Frederic D. Highland, US 4,924,408) in view of Luke (Edward Allen Luke, "A Rule-Based specification system for computational fluid dynamic").

Claim 3:

Highland discloses the method of claim 1, but does not explicitly disclose the processing comprises multi-pass execution when there are logical loops. However, Luke in the same analogous art of rule-based system discloses a way of treatment of recursive dependencies (see for example, p.39, lines 19-21, "Actually, a recursion loop in the graph represents iteration over entities, and so a loop of this form must be repeatedly evaluated until all attributes have been generated."). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to schedule multi-pass execution when there is a logical loop as once suggested by Luke. One would have been motivated to combine Luke's and Highland's methods together to traverse the entire network for the propagation of results as suggested by Highland. (see for example, col.5, lines 11-15, "a mechanism must be developed to compile the traversal of the network for the propagation of results in order to preserve the dynamic nature of the knowledge based system")

Claim 4:

Highland and Luke disclose the method of claim 3 above. Luke further discloses that the processing comprises providing an endless loop terminating condition (see for example, p.28, lines 15-18, "the loop termination condition"). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to further include Luke's teachings in Highland and Luke's

method in order to terminate endless loop as once suggested by Luke (see for example, p.40, lines 8-9, "In order to remove this loop...")

14. Claims 26-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Highland (Frederic D. Highland, US 4,924,408)

Claims 26-28:

Claims 26-28 are the computer program products, disposed on a computer readable medium for business processing automation, which are the product version of the claimed methods discussed as in claims 1, 5 and 6 above respectively. It is well known in the computer art to practice and store the computer readable code in such computer readable storage medium. Therefore, these claims are also obvious over Highland.

15. Claims 7-15 and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Highland (Frederic D. Highland, US 4,924,408) in view of Burke (Burke et al., US 5,423,041).

Claim 7:

Highland discloses the method of claim 6 to generate the dependency graph, but does not disclose resolving logical conflicts using override instructions. However, Burke in the same analogous art of rule-based system discloses a method for assigning each rule a priority to solve the conflict problem. (see for

example, col.3, lines 33-35, "...an application programmer may assign each rule a priority. If multiple rules are satisfied at the same time, the rule with the highest priority is selected for firing."). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to assign a priority for each rule in Highland's graph (network). One would have been motivated to do so to solve conflict problem in Highland's system by using the Burke's suggestion above to override low priority rule while two rules conflict.

Claim 8:

Highland and Burke disclose the method as in claim 7 above. Highland further discloses the dependency graph further comprises analyzing the rule set with a business logic generation utility optimized for one of a plurality of target programming languages and generating optimized logic for a selected target programming language (see for example, col.7, lines 25-38, "The implementation of rules and the inference engine as procedural, program code provides additional efficiencies...", also see col.8, lines 1-9, "It should also be noted that the program code generated using the compilation technique according to the present invention includes the logic for the inference engines...").

Claim 10:

Highland and Burke disclose the method as in claim 8 above, Burke further discloses the target programming language is C++ (see for example, col.2, lines

46-68, "Although this description is in terms of C++, the basic concepts of the invention may be used with other object-oriented programming languages"). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to use C++ as target programming language to implement Highland and Burke's method. One would have been motivated to do so to take advantages of object-oriented language as once suggested by Burke (see for example, col.2, lines 51-59, "'permits the rule system to operate directly on class instances created with object-oriented language, also see, col2., line 67, "inheritance of object attributes")

Claims 9 and 11-14:

Highland and Burke disclose the method as in claim 8 above, Burke further discloses the target programming language may be used with other object-oriented programming languages (see for example, col.2, lines 46-68, "Although this description is in terms of C++, the basic concepts of the invention may be used with other object-oriented programming languages"). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to use Java, JavaScript, Jython, Visual Basic or C# to do object-oriented programming to implement Highland and Burke's method. One would have been motivated to do so to take advantages of object-oriented language as once suggested by Burke (see for example, col.2, lines 51-59, "'permits the rule system to operate directly on class instances created with object-oriented

language, also see, col2., line 67, "inheritance of object attributes")

Claim 15:

Highland and Burke disclose the method as in claim 8 above, Burke further discloses the business logic generation utility's generated processing logic comprises a series of calls to a working memory database to retrieve, manipulate and update data (see for example, col.3, lines 59-61, "During its match phase, the inference engine tests each rule's premise against the current working memory", also see col.4, lines 10-12, "These actions may update the working memory by creating new objects and removing old objects."). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to further combine Burke's new feature to Highland and Burke's method as discussed in claim 8 before. One would have been motivated to do so to make the system more efficient as once indicated by Burke (see for example, col.2, lines 14-15, "directly access the user-defined object via the working memory").

Claim 29:

Claim 29 is the computer program product, disposed on a computer readable medium for business processing automation, which is the product version of the claimed method discussed as in claim 7 above. It is well known in the computer

art to practice and store the computer readable code in such computer readable storage medium. Therefore, this claim is also obvious over Highland and Burke.

16. Claims 19 and 21-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Burke (Burke et al., US 5,423,041)

Claims 19 and 21-24:

Burke discloses the method of claim 18, in which the target programming language may be used with other object-oriented programming languages (see for example, col.2, lines 46-68, "Although this description is in terms of C++, the basic concepts of the invention may be used with other object-oriented programming languages"). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to use Java, JavaScript, Jython, Visual Basic or C# to do object-oriented programming to implement Burke's method. One would have been motivated to do so to take advantages of object-oriented language as once suggested by Burke (see for example, col.2, lines 51-59, "permits the rule system to operate directly on class instances created with object-oriented language", also see, col2., line 67, "inheritance of object attributes").

***Conclusion***

17. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.
18. Applicant's arguments with respect to claims rejection have been considered but are moot in view of the new grounds of rejection.

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

19. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Zheng Wei whose telephone number is (571) 270-1059 and Fax number is (571) 270-02059. The examiner can normally be reached on Monday-Thursday 8:00-15:00.

Art Unit: 2192

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature of relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571- 272-1000.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

ZW

  
TUAN DAM  
SUPERVISORY PATENT EXAMINER